School of Computer Science and Informatics

Coursework Submission Cover Sheet

Please use Adobe Reader to complete this form. Other applications may cause incompatibility issues.

Student Number	1521603
Module Code	CM3203
Submission date	02/02/2020
Hours spent on this exercise	20

Special Provision

(Please place an x is the box above if you have provided appropriate evidence of need to the Disability & Dyslexia Service and have requested this adjustment).

Group Submission

For group submissions, *each member of the group must submit a copy of the coversheet.* Please include the student number of the group member tasked with submitting the assignment.

Student number of submitting group member

By submitting this cover sheet you are confirming that the submission has been checked, and that the submitted files are final and complete.

Declaration

By submitting this cover sheet you are accepting the terms of the following declaration.

I hereby declare that the attached submission (or my contribution to it in the case of group submissions) is all my own work, that it has not previously been submitted for assessment and that I have not knowingly allowed it to be copied by another student. I understand that deceiving or attempting to deceive examiners by passing off the work of another writer, as one's own is plagiarism. I also understand that plagiarising another's work or knowingly allowing another student to plagiarise from my work is against the University regulations and that doing so will result in loss of marks and possible disciplinary proceedings.

Initial Project Plan

A Dashboard Interface for Policing Large Events

1   Project Description

Cities often host large events. In recent years, for example, Cardiff has played host to diverse events including the 2014 international NATO Summit and the 2016 UEFA Champions League Final, as well as numerous smaller events requiring extensive police and security operations. Policing such events requires good situational awareness (SA): an ability to keep track of what is happening in real-time. Effective SA in this scenario requires the use of data collected in real-time from multiple sources (for example, CCTV, car number-plate recognition systems, automatic facial recognition) along with any reports from officers stationed around the city.

 The Crime & Security Research Institute at Cardiff is doing research in SA using artificial intelligence (AI) and machine learning (ML) tools that can process information feeds such as video, audio and text. ("A pilot study on detecting violence in videos fusing proxy models.,", 2019) [1] is a good study for reference in understanding what situations these tools might process, specifically in the detection of violence in CCTV feeds. The aim of this project is to build a dashboard in the form of a Web interface that can integrate multiple tools developed by the Institute to produce a "concept demonstrator" of how AI, ML and human-computer interaction (HCI) can support SA for large events in a city. The interface will link the multiple tools and their pre-generated "feeds" into a map-based interactive visualisation [Figure 1], to demonstrate the feasibility and use of such a system to an audience of police partners and potentially other agencies concerned with the security of urban areas.

Other currently available implementations for similar problems are specific to a particular application, such as the conversational interface for tracking cars in a city [2], so there is a distinct lack of existing pluggable framework solutions for this kind of situational awareness. As such the final deliverable should be a pluggable framework, open to a wide range of sensor types, and markable information, made publicly available on GitHub, to fill this gap. The final deliverable also needs to be available offline, so demonstrations can be run without relying on an Internet connection.

The end result of the implementation should be a website that can gather information on sensors and events from AI/ML tools, and use the location stored within these to add the sensors and events to an interactive map. Users should then be able to view information on these map markers, as in Figure 2, and report information back to the system for further data interpretation by an intelligence analyst.
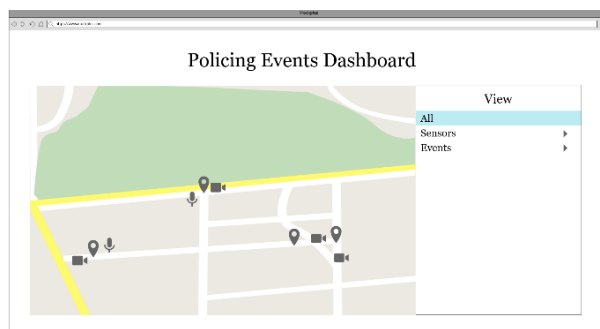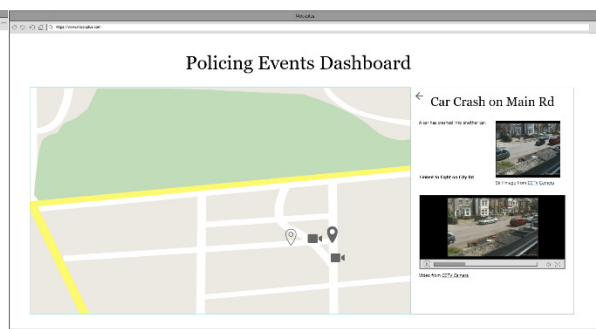

Figure 1 – Wireframe Design of Interactive Map Page   *Figure 2 – Wireframe Design of Marker Description Page*

2   Project Aims and Objectives

2.1   Primary (MUST)

1. Have a system to store static pre-recorded and tool-analysed images and video/ audio clips, along with a location and other relevant information for map markers
2. Have an interactive map on a web interface marking locations of sensors and events
    -   "sensor" markers on the map linked to the geolocation of a captured image/ video or audio clip (e.g. the location of a CCTV camera that recorded some relevant footage)
    -   "event" markers showing the location of events captured by the sensors
3. Display a list on the web interface of all the sensors and events currently marked on the map
4. Allow the list of events and sensors to be sorted/ filtered as such:
    -   Sensors: sorted by time, filtered by type (camera/ person/ etc)
    -   Events: sorted by time or priority, filtered by type (fight/ car crash/ etc) or "has related events"
5. Allow a user to access detailed information on a sensor or event by clicking on its marker on the map or its name on the list; detailed information defined as below:
    -   Sensors – captured images/ video or audio clips, and any saliency mapping on those, with an analytical description from the AI/ML tool
    -   Events – any related images/ video or audio clips from sensors, with saliency mapping if it exists, and semantic descriptions from the AI/ML tool
6. Store times with each sensor/event, allowing demonstrations to appear to have an active "timeline", with new events appearing as the demonstration progresses
    -   Should allow for both a general timeline: sensors/events brought to map based on the stored time and the seconds since the start of the demonstration, and a "skip-based" timeline: allowing the demonstrator to "skip" to the next or return to the previous relevant time (appearance of an event)
7. Be available offline for demonstrations
8. Be publicly available via GitHub

2.2   Secondary (SHOULD)

9. Be able to show "complex events"
    -   Complex Events – two or more related events, which could be marked as related by AI (based on type and proximity; logical reasoning) or humans
10. Allow reporting back from users by sending information to Sketch
    -   All marker information along with an optional note from the user should be sent to Sketch
    -   Sketch – a tool under development by IBM and the Crime and Security Research Institute which allows an intelligence analyst to interpret data (as described in [3])
11. Allow complex events to be exported to Sketch
12. Have a live system to link the AI/ML tools and their outputs, allowing the tools to output information which can then be sent to and displayed on the web interface
13. Show pre-generated geolocated Twitter posts as markers on map

2.3   Tertiary (COULD)

14. Allow a mobile user to submit an event description of their own, which would be shown on the interactive map as a human sensor
15. Allow human users to mark two or more events as related, making them complex events
16. Link Twitter post markers on map to an event, using key words and location

3   Ethics

No ethics certification or training is needed for this project, as the Twitter post will have been pre-generated by the Crime and Security Research Institute, so won't have any relation with any actual person or persons, thus not requiring the storage of any personal information such as names or locations. Any evaluation from persons other than myself or the project supervisor, will be researchers or research collaborators from the Crime and Security Research Institute. At no point will a general member of the public be asked to fill out a survey or give a review of the product.

4   Work Plan

4.1   Task Description

Task Due Date	Description
Deliverables	
Task 1 03/02/2020	Initial Project Plan Deadline
Task 2 07/05/2020	Final Project Report Deadline
Implementation	
Task 1 09/02/2020	Set up system to store information about AI/ML tool events locally. Linked to Objectives 1 and 7. Planning to use JSON documents to store the information, and NodeJS to store these along with images, videos, and audio. Will use self-generated test data at this stage.
Task 2 11/02/2020	Set up locally hosted map system. Linked to Objectives 2 and 7. Planning to use LeafletJS with OpenStreetMap tiles for interactive map, so will need to set up locally hosted OpenStreetMap server (only covering a map of Cardiff at this stage).
Task 3 13/02/2020	Build a website which displays the interactive map as above with markers based on locations stored in the JSON documents, should also have a list of map markers on the side. Upload to GitHub, and commit changes regularly. Linked to Objectives 2, 3, and 8. Planning to use the NodeJS API to link the website to the NodeJS server, so information for markers can be collected.
Task 4 18/02/2020	Gather images and video/ audio clips, and store these locally with relevant time and location information. Linked to Objectives 1 and 6. Planning to build these into JSON Documents as in Task 1, the time will be stored as seconds since start of demonstration, and a skip-based event number.

Task 5 21/02/2020	Add ability to view detailed information on a sensor/ event in a sidebar on the website. Linked to Objective 5. Planning to GET images, videos, and audio clips from NodeJS. May have them running on NodeJS system to appear live.
Task 6 25/02/2020	Create Sort/ Filter functionality for the list of map markers. Linked to Objective 4.
Task 7 29/02/2020	Create a general timeline for demonstrations. Linked to Objective 6. Planning to send API GET requests with seconds since demonstration start, to only collect events and sensors with time before request.
Task 8 03/03/2020	Create a skip-based timeline for demonstrations. Linked to Objective 6. Planning to send API GET requests with event point, retrieving all events and sensors for that point and refreshing the page to remove some in case of skip backwards.
Task 9 09/03/2020	Add compatibility for complex events. Linked to Objective 9. Planning to add a field in the JSON documents to allow a related event to be linked.
Task 10 20/03/2020	Add functionality for reporting back, by exporting information to Sketch. Linked to Objectives 10 and 11. Planning to add an "ignore" field to the JSON document if a marker is deleted. Sketch accepts JSON documents as input, so exporting the information could be done with only minor format adjustments.
Task 11 27/03/2020	Create a live system to collect information from AI/ML tools and update the website with it. Linked to Objective 12. Planning to use NodeRed to create a flow between the tools and the web interface. This will require the tools to generate correctly formatted JSON documents, or a new system to automatically generate them from randomised information.
Task 12 31/03/2020	Add pre-generated Twitter posts to map. Linked to Objective 13. Planning to add these as JSON documents with "Twitter Post" sensor type.
Task 13 10/04/2020	Create a mobile interface allowing entering of information from user, should allow inputting of new event with description, adding description to current event, and creating a link between 2 or more events. Linked to Objectives 14 and 15. Planning to allow JSON documents to be created with Human sensor information, events to be edited with Human sensor information, and events to be edited with new links to related events.
Task 14 14/04/2020	Add a system to automatically link a Twitter post to an event based on key words, proximity, and human entry from Task 13. Linked to Objective 16.

1521603

Meetings	
Meeting 1 30/01/2020	Discussion About the Initial Project Plan
Meeting 2 04/02/2020	First Implementation Meeting with Crime and Security Research Institute AI/ML Team
Meeting 3 03/03/2020	Review of Implementation Progress with Focus on MUST Objectives
Meeting 4 31/03/2020	Review of Implementation Progress with Focus on SHOULD Objectives
Meeting 5 14/04/2020	Review of Implementation Progress with Focus on COULD Objectives
Documentation	
Task 1 03/02/2020	Produce Initial Project Plan
Task 2 09/02/2020	Final Report - Introduction
Task 3 17/02/2020	Final Report - Background
Task 4 02/03/2020	Final Report - Approach
Task 5 16/03/2020	Final Report - Implementation
Task 6 03/04/2020	Final Report - Results and Evaluation
Task 7 10/04/2020	Final Report - Future
Task 8 20/04/2020	Final Report - Conclusions
Task 9 28/04/2020	Final Report - Reflection
Task 10 07/05/2020	Final Report - Clean Up

4.2   Gantt Chart
Figure 3 – Gantt Chart

5   Works Cited

[1] M. R. Vilamala, L. Hiley, Y. Hicks, A. Preece and F. Cerutti, "A pilot study on detecting violence in videos fusing proxy models.," 2-5 July 2019. [Online]. Available: http://orca.cf.ac.uk/id/eprint/122640.

[2] D. Harborne, D. Braines, A. Preece and R. Rzepka, "Conversational control interface to facilitate situational understanding in a city surveillance setting," 13 July 2018. [Online]. Available: http://orca.cf.ac.uk/id/eprint/113152.

[3] D. Braines, J. Ibbotson, D. Shaw and A. D. Preece, "Building a "living database" for human-machine intelligence analysis.," 6-9 July 2015. [Online]. Available: http://orca.cf.ac.uk/id/eprint/76333.
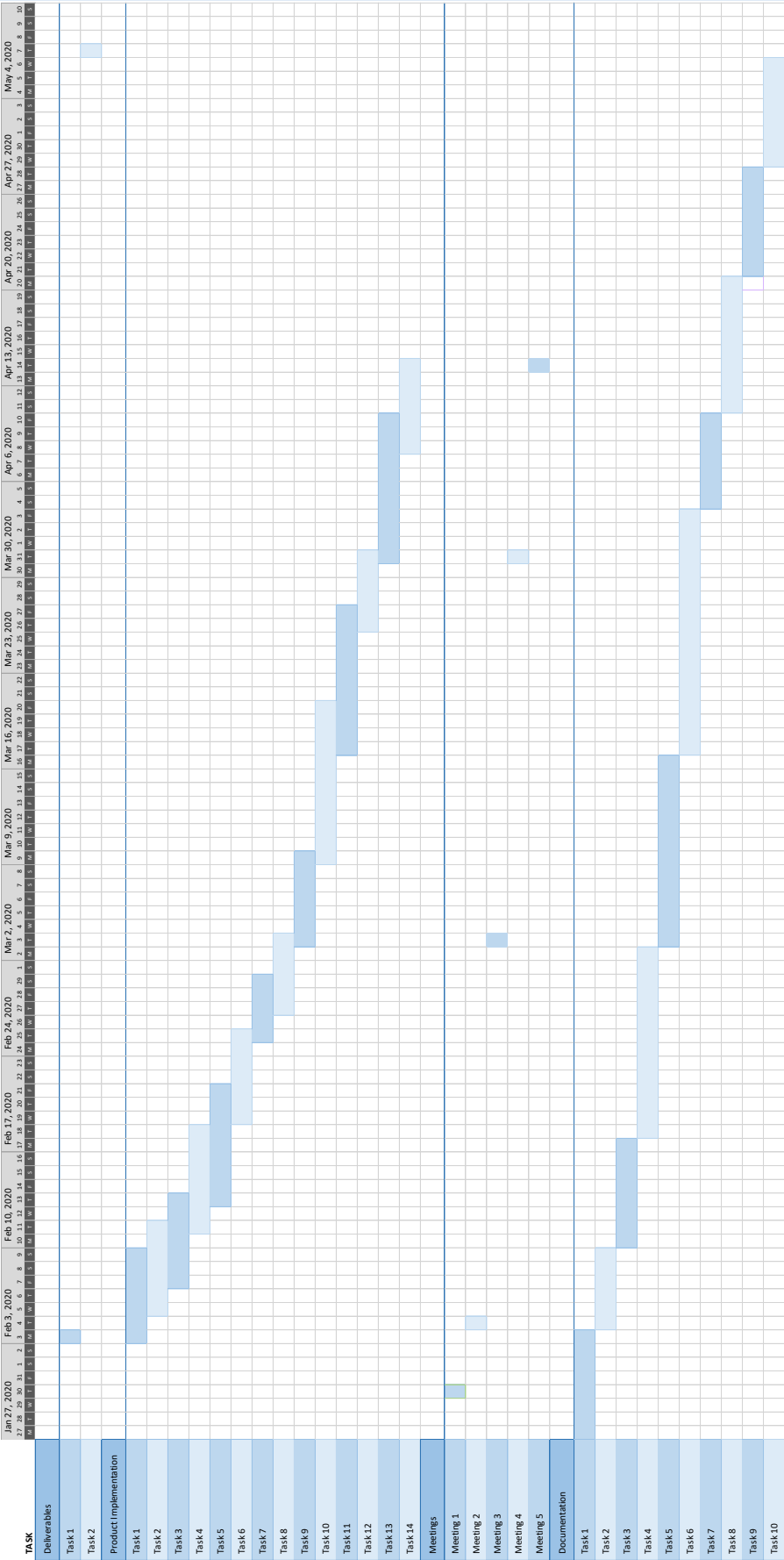
1521603

6   Appendix

Figure 3 – Gantt Chart